21555 Non-Transparent PCI to PCI Bridge

Bridges Product Operation
Intel

Intel Labs

Legal Disclaimer

Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

This document contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product is available. Verify with your local sales office that you have the latest datasheet before finalizing a design.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling1-800-548-4725, or by visiting Intel's website at http://www.intel.com.

*Third-party brands and names are the property of their respective owners

Intel Labs

Agenda

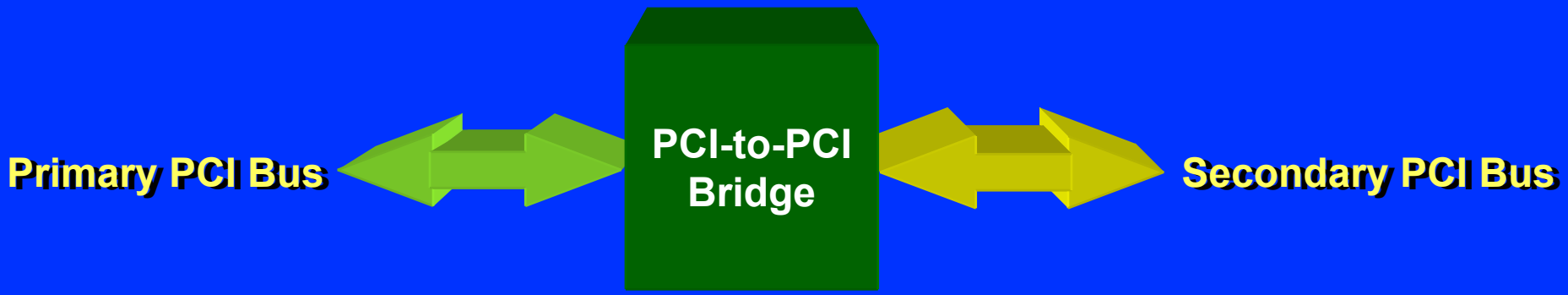
- What is a PCI to PCI bridge?
- Transparent and Non-transparent Bridges
- The *2155x Non-transparent* Bridge
- New Features
- Bridge Products Roadmap

Intel Labs

What is a PCI-to-PCI Bridge?

Intel Labs

PCI-to-PCI Bridges Attach to a PCI Bus and Create a Second, Independent PCI Bus

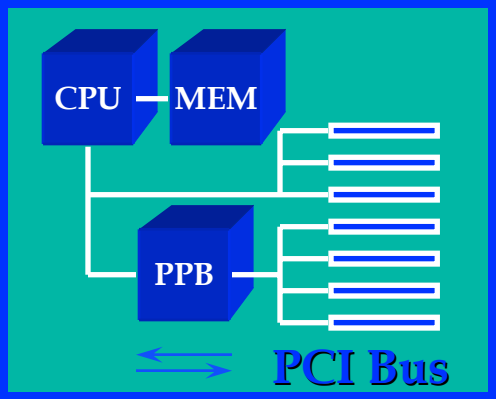


Intel Labs

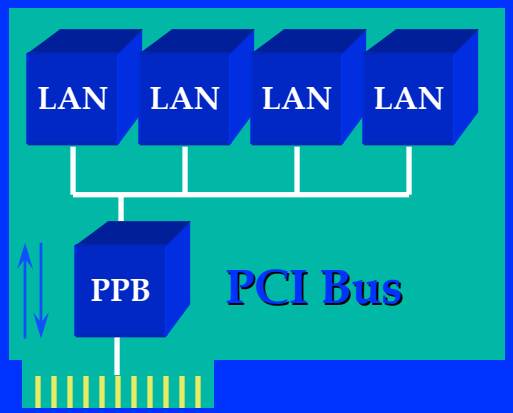
PCI to PCI Bridge

- Extends a bus
  - more PCI slots on a motherboard
  - enables multi-function PCI option cards
- Bus Segments run concurrently
  - higher system throughput with lower latency

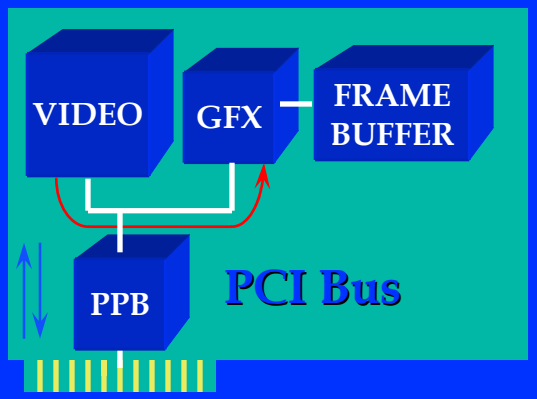
Motherboard Slot Expansion



Multi-Device PCI Option Card

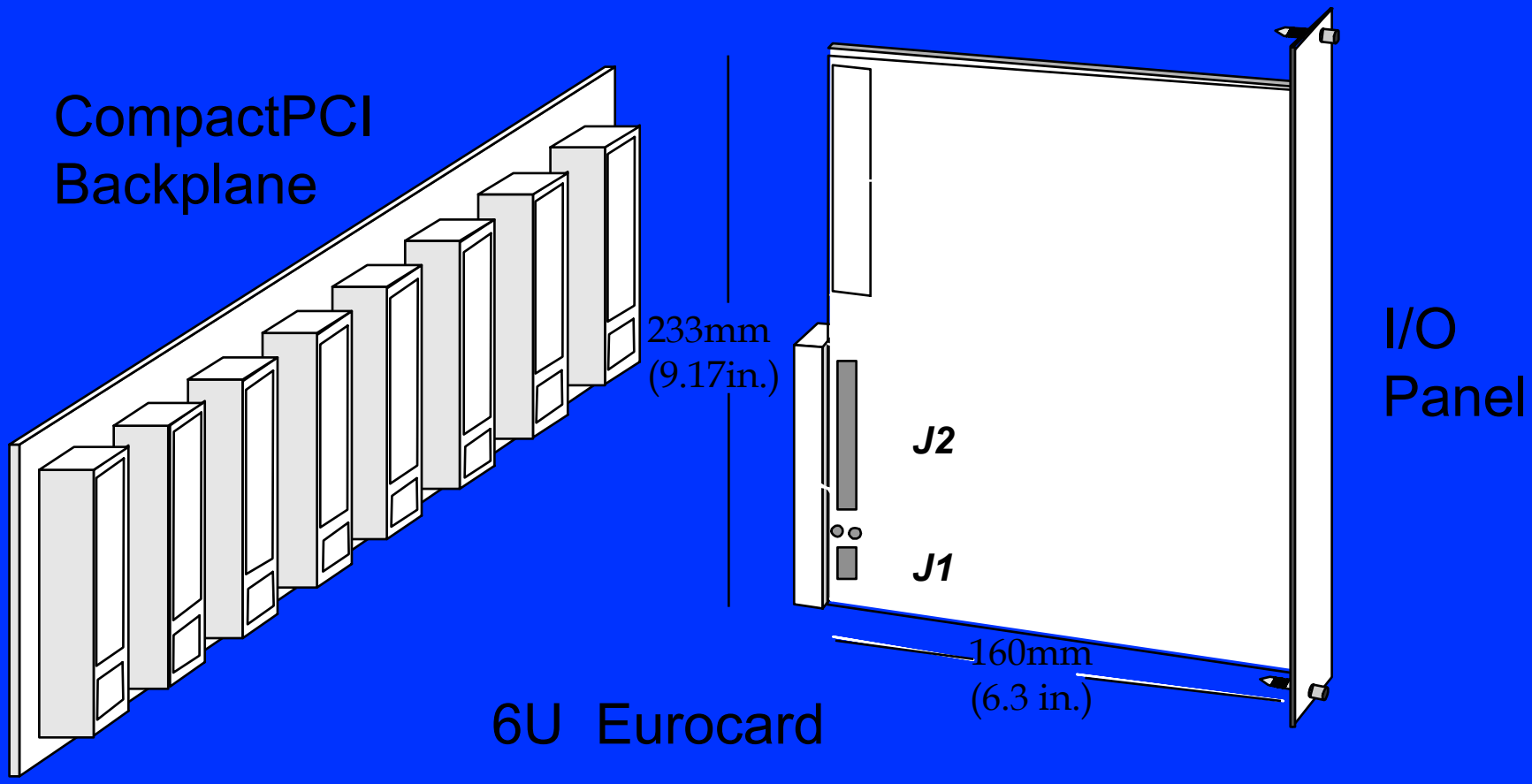


Multi-Function PCI Option Card



Intel Labs

CompactPCI®



Intel Labs

Transparent PPB
Host Configuration

- ***All*** devices within the hierarchy appear as individual PCI devices
  - Boundaries of the subsystem are difficult to determine
  - Host will typically load standard device drivers for each component found

Difficult to load a driver which controls subsystem as a whole!

Intel Labs

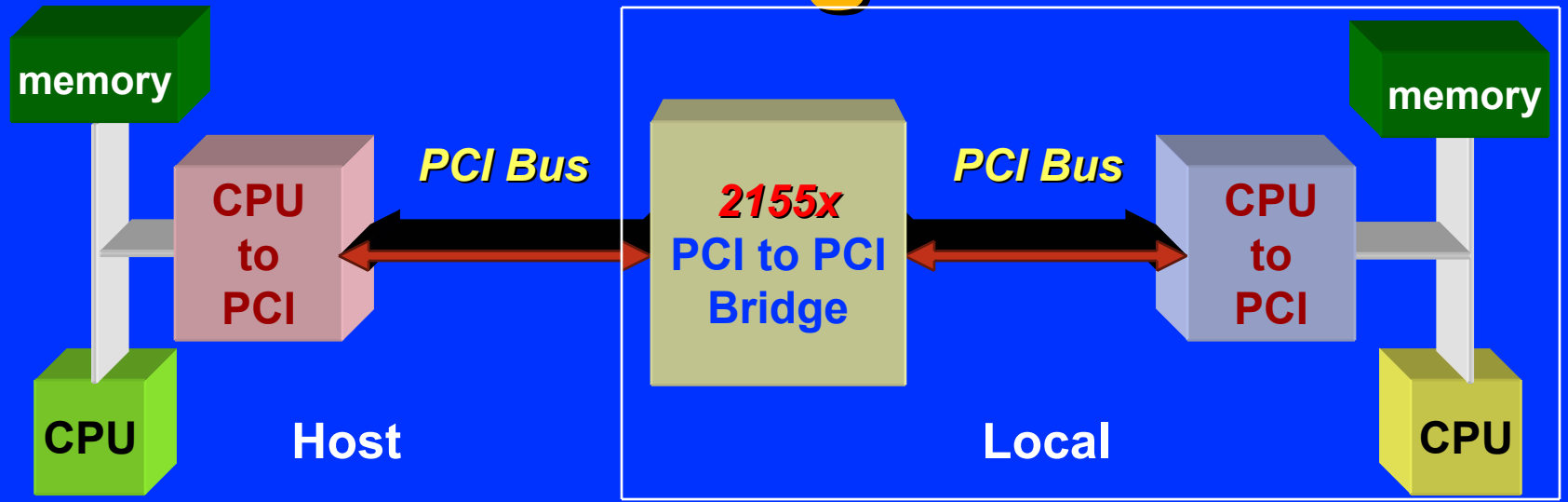
Non-transparent PCI to PCI Bridge

- ***Similar*** **in function to a standard PCI-to-PCI bridge**
  - **–Isolates the loading of additional PCI devices**
  - **–Allows concurrent operation of both PCI bus segments**

But it has some important differences...

Intel Labs

Non-Transparent PCI to PCI Bridge



Specifically designed to bridge between two intelligent domains

- Secondary CPU
- Independent PCI clocks
- Independent PCI address spaces
- Address translation between PCI buses

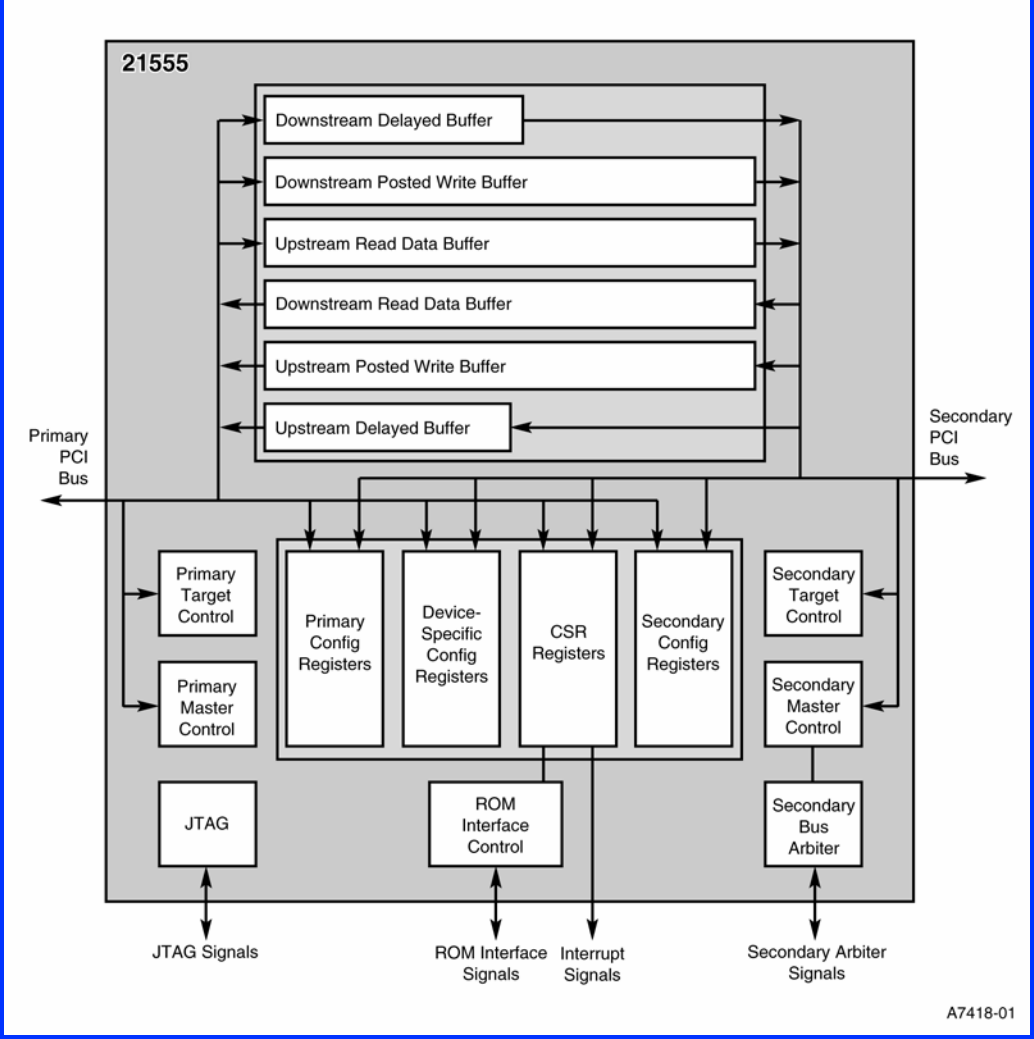
Intel Labs

Non-Transparent PCI to PCI Bridge

- Presents the sub-system as a single “device” to the host
  - allows loading of a single driver for the entire sub-system
  - local processor can independently configure and control the local sub-system
- Hides sub-system resources from host
  - address translation resolves resource conflicts between the host and local sub-systems
- Allows independent primary and secondary bus clocking

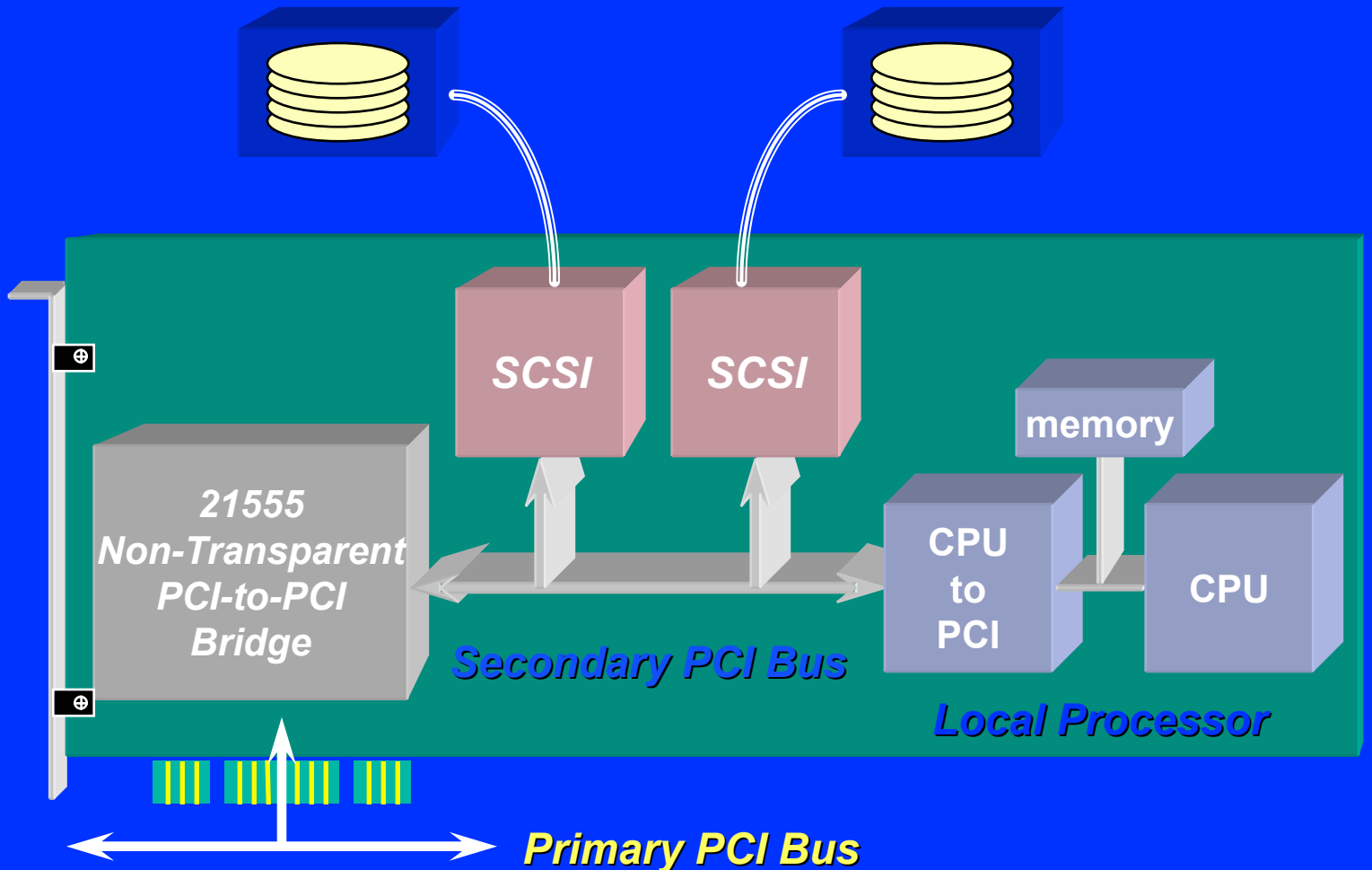
Intel Labs

21555 Microarchitecture



Intel Labs

Non-Transparent PPB Application: RAID Controller



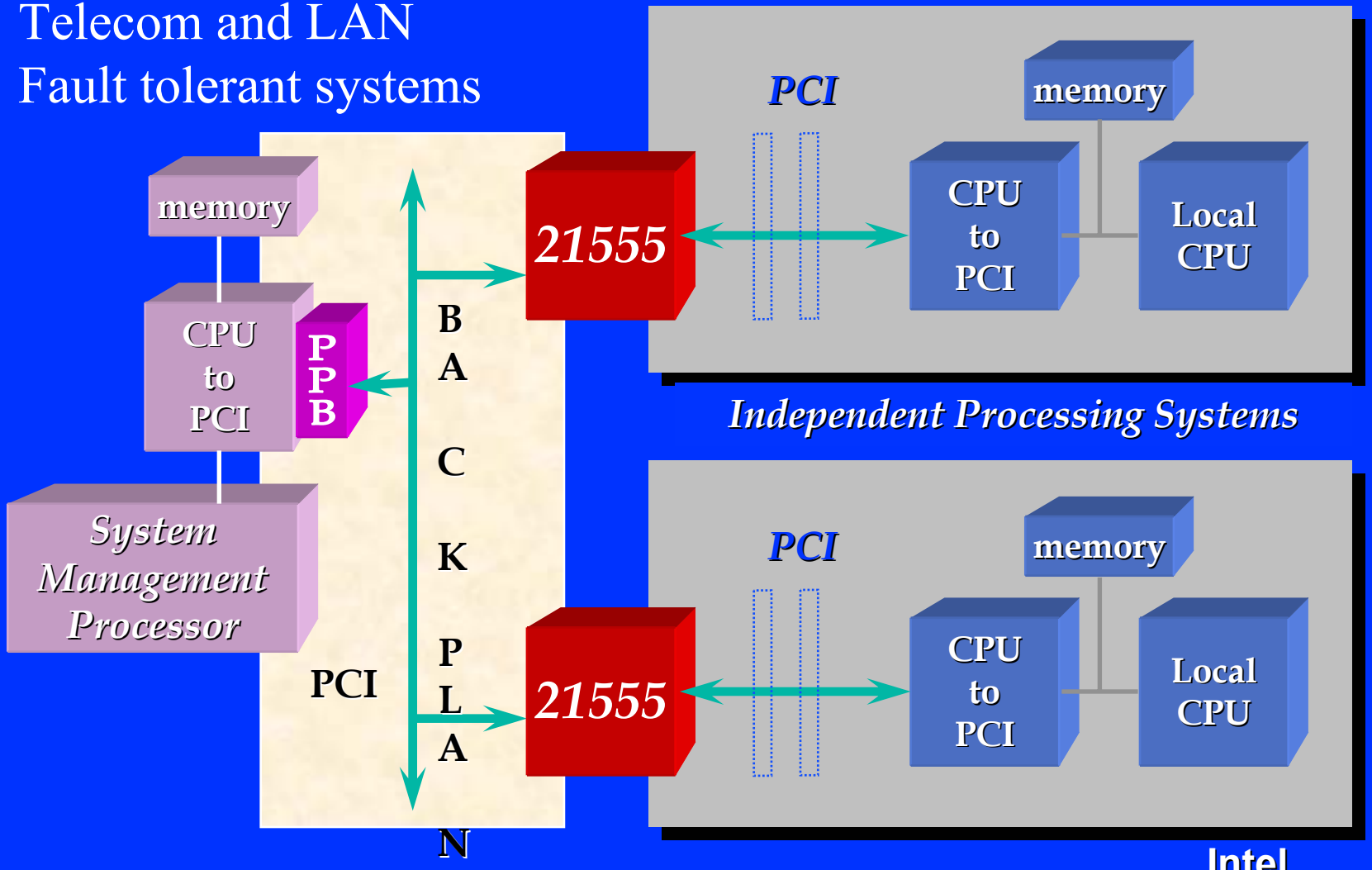
Intel Labs

Non-Transparent PPB

Backplanes/Gateways

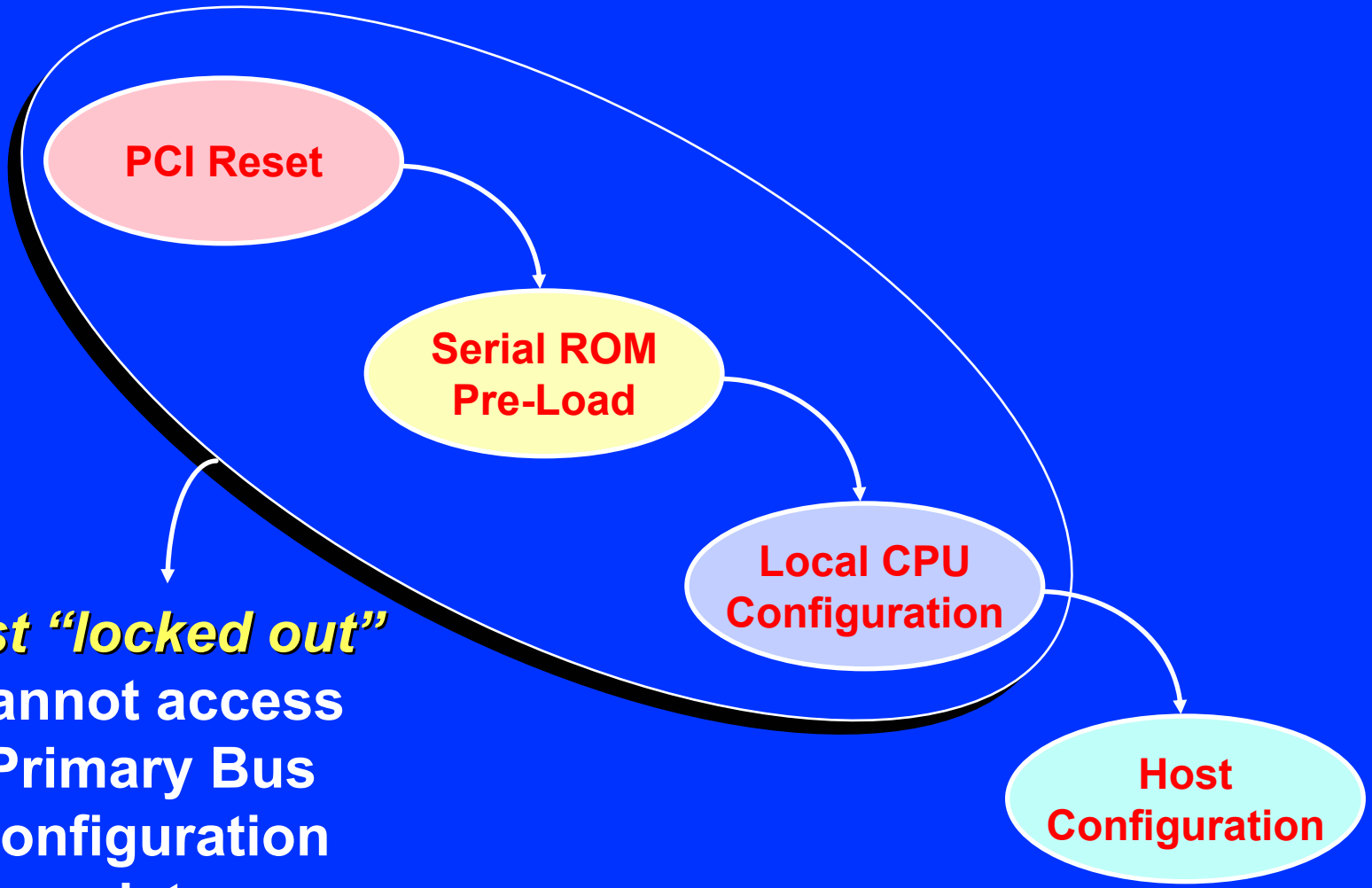
Telecom and LAN

Fault tolerant systems



Intel Labs

Non-Transparent PPB
Initialization Sequence



Host "locked out" cannot access Primary Bus configuration registers

Intel Labs

Non-Transparent PPB
SROM Initialization

- Following PCI reset the *2155x's* initial state is loaded from a serial ROM
  - Vendor specific identification
  - Base address register setup information
  - Initialization status bit
    - Locks out host from access of configuration registers

Intel Labs

Non-Transparent PPB
Local CPU Initialization

- Local CPU configures secondary bus devices

  Address assignments of secondary bus devices are done by the local CPU via standard PCI configuration registers

Non-Transparent PPB
Local CPU Initialization

- ***Non-transparent bridges*** **use a type 0 configuration header**
  - **Unique secondary-side set of configuration registers**
    - **Identifies the non-transparent bridge to local CPU**
  - **Base address registers for the CSRs**
    - **Control and status, doorbell, mailbox, and I_2O® messaging registers**
    - **Indirect access of primary interface configuration information**

I_2O is a registered trademark of the I_2O ® Special Interest Group

Intel Labs

Non-Transparent PPB
Local CPU Initialization *continued...*

- Base address registers for access of primary bus resources
  - E.G. A buffer in the host CPUs memory

- Local CPU can override the *2155x's* initial state
  - Sets initialization status bit when finished
    - Enables host access of configuration registers (ends host lockout)

Intel Labs

Non-Transparent PPB
Host Configuration

2155x uses a type 0 PCI configuration header

- Does not use PCI bus hierarchical Configuration code
  - Host configures only the *NTB*

The host is unaware of any other components behind the bridge

Intel Labs

Driver Initialization

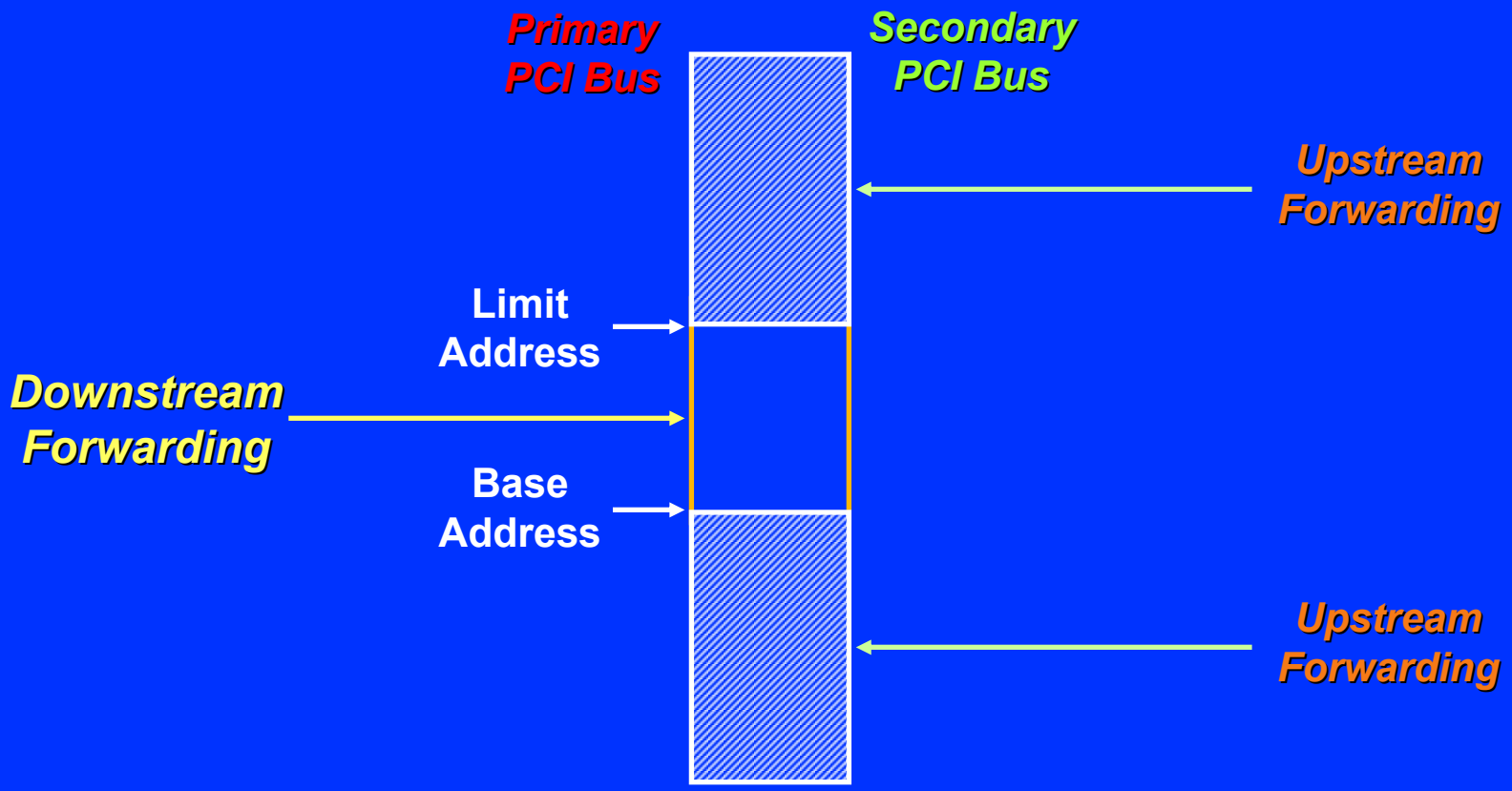
- *It is possible* to implement a *2155x* design that does not use either a serial or local processor preload
  - Only the reset values of the read only registers are used
  - All forwarding BAR's are disabled
  - Configuration registers are accessible
  - CSR registers can be mapped into memory
  - A parallel ROM can be accessed via the CSR mechanism
  - $I_2O^{®}$ message unit, doorbell and scratchpad registers are accessible
  - *2155x* configuration registers can be accessed using a downstream indirect configuration mechanism

Intel Labs

Non-Transparent PPB
Host Configuration

- One set of PCI configuration registers for the entire subsystem
  - Vendor specific identification information
  - Base address registers for the CSR's
  - Base address registers for access of secondary bus resources
  - Expansion ROM base address register

Transparent PPB
Transaction Forwarding



Three ranges supported:

I/O, memory, Prefetchable memory

Intel Labs

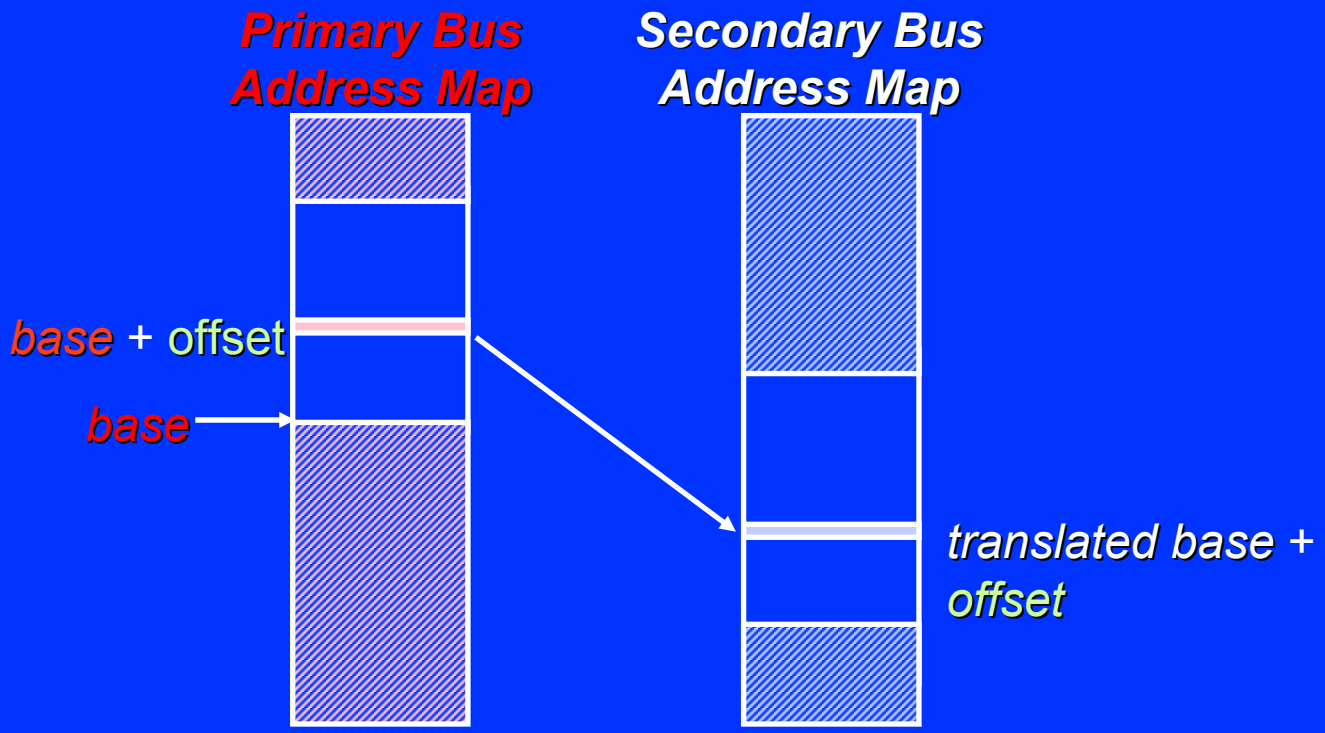
Transparent PPB
Transaction Forwarding Issues

- Addresses on secondary PCI bus are assigned by host
  - Host address assignments may conflict with the needs of the local CPU
- All secondary bus devices can be accessed from the primary interface
  - Subsystem resources may need to be isolated from host access

Intel Labs

Non-Transparent PPB

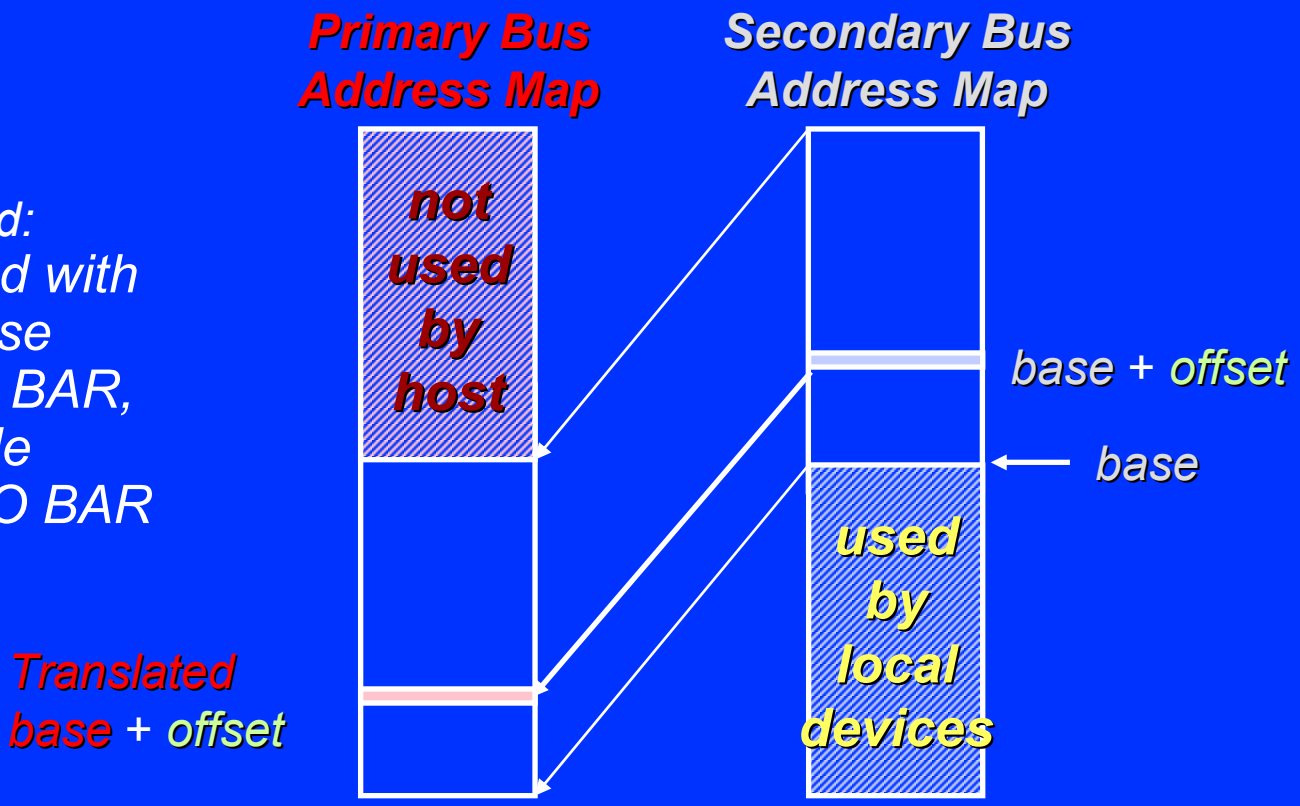
Downstream Transaction Forwarding



- Direct offset method
  - Base replaced with translated base
  - Two memory bars, one selectable memory or I/O BAR

21555 PPB
Upstream Transaction Forwarding

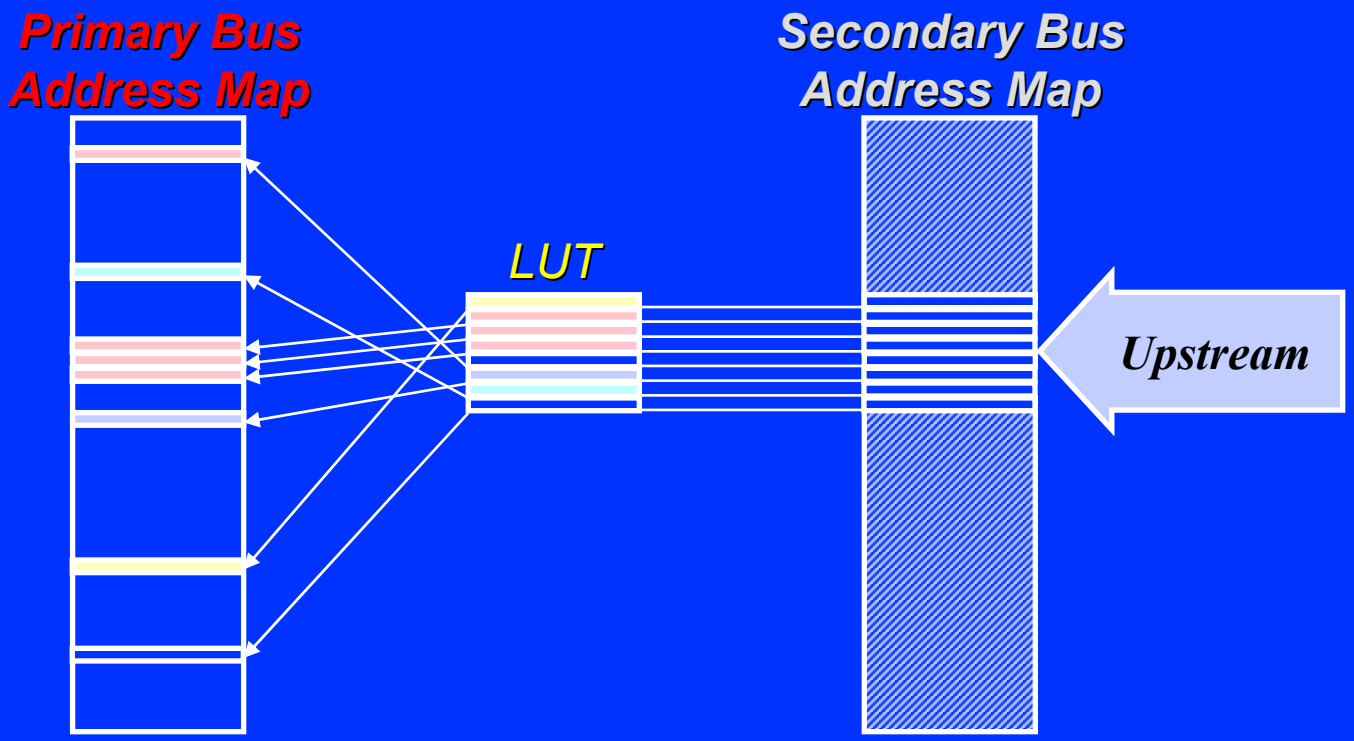
Direct offset method:
Base replaced with translated base
One memory BAR, one selectable memory or I/O BAR



- **Simple method to resolve address conflicts**
  - **System memory of 4 Gbytes or less**

Intel Labs

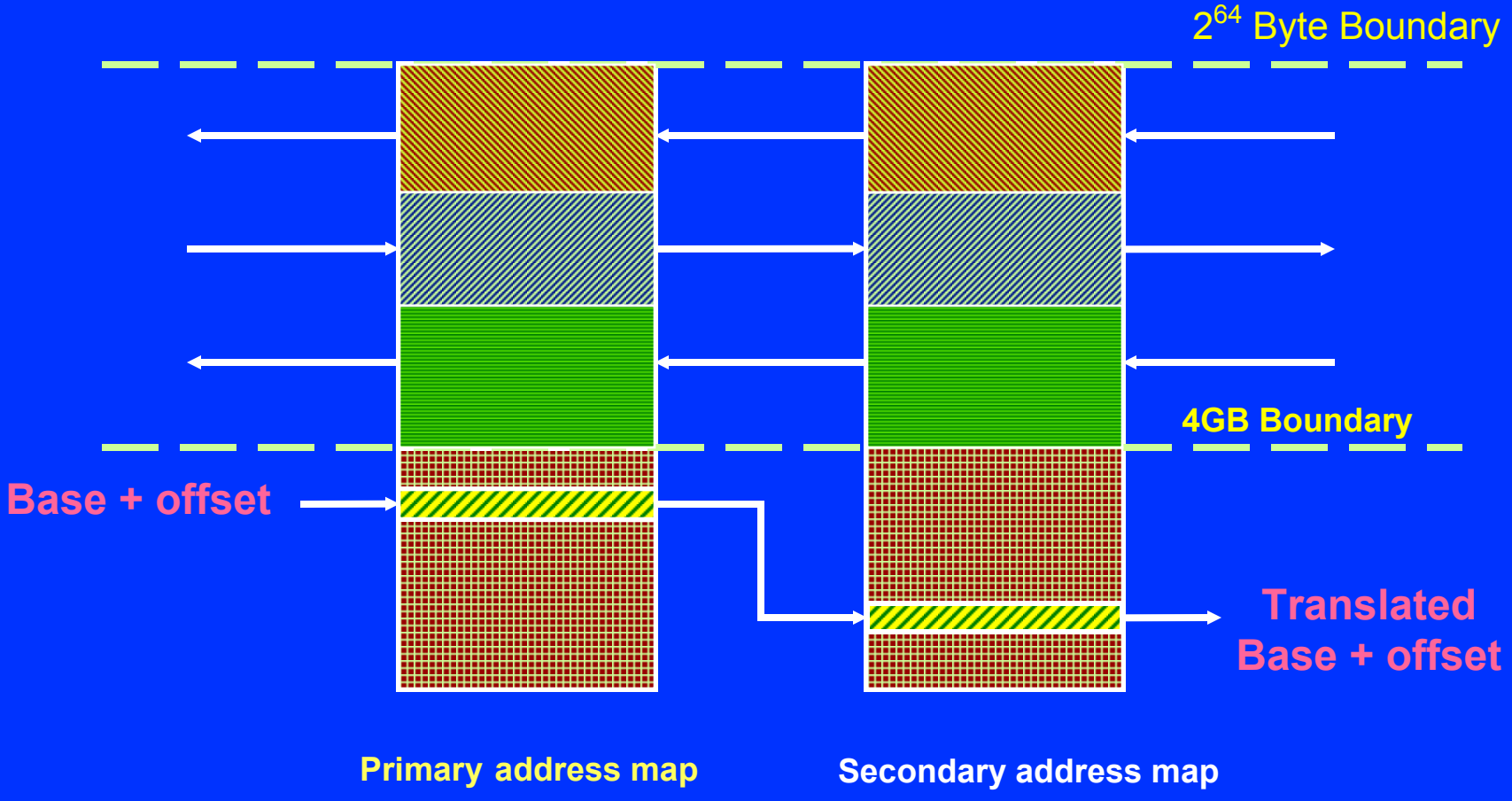
21555 PPB
Upstream Transaction Forwarding



- Look up table (LUT) method
  - 64 entry table provided for one *upstream* memory BAR
  - Translation of base address using memory page granularity LUT

Intel Labs

Dual-Address Transaction Forwarding



Intel Labs

Subtractive Decode

- I/O transactions can be subtractively decoded by the ***NTB***.
  - Enabled by setting a configuration bit
  - Can be enabled ***either*** downstream or upstream, but ***not*** both
  - Only one subtractive decode agent allowed on ***any*** PCI bus segment

Intel Labs

Indirect Configuration Transactions

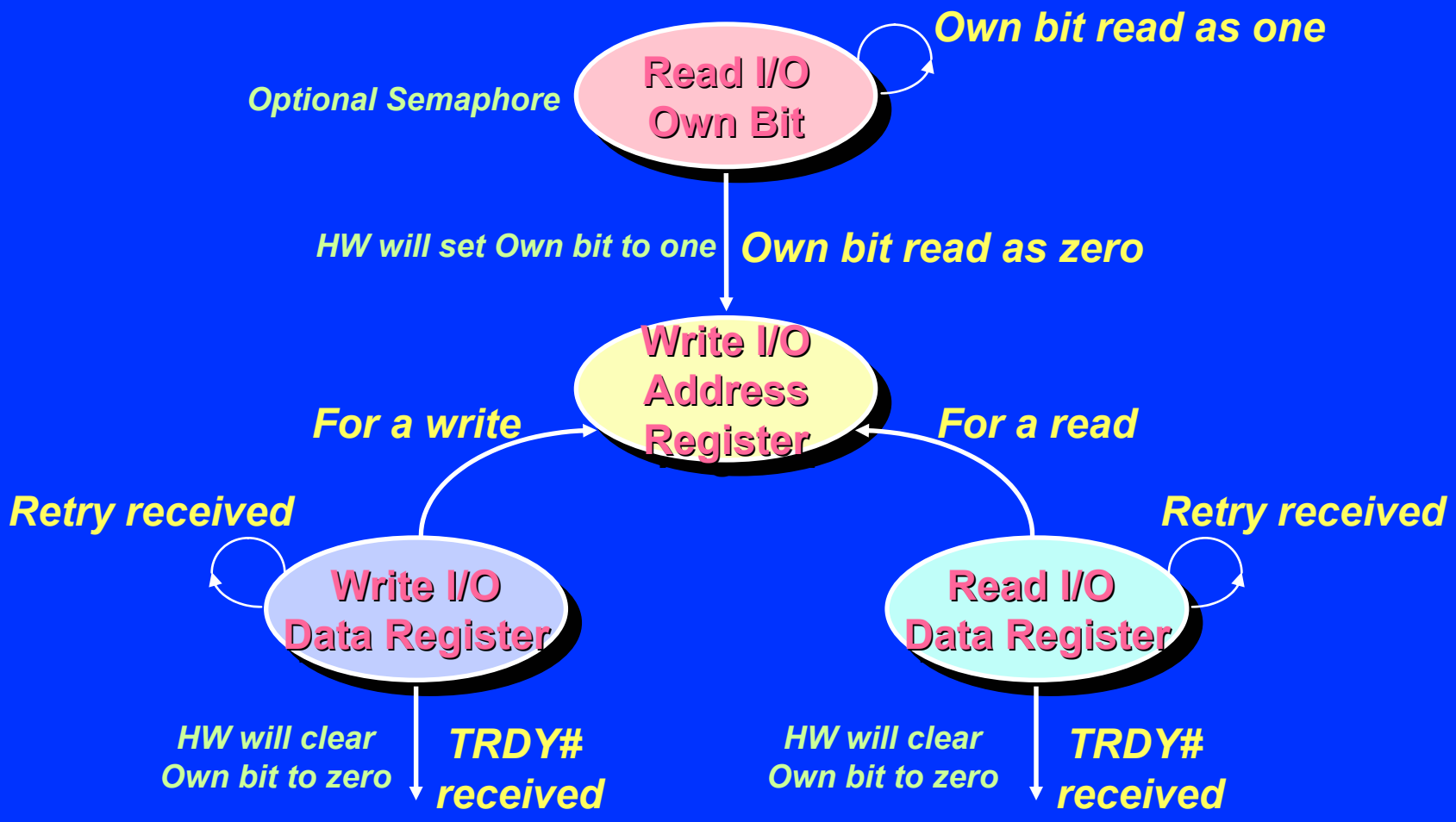
- *2155x* does *not* support *Transparent PCI-to-PCI bridge* hierarchical configuration mechanisms
  - Host configures primary bus devices
  - Local processor configures secondary bus devices
- *2155x does* provide an alternative method.
  - Can initiate type 0 or type 1 configuration transactions via CSR accesses
- CSR indirect configuration method is the same as for I/O
  - Indirect configuration address and configuration data registers
    - Mapped in both configuration space and I/O space
  - Access completed as a delayed transaction

Intel Labs

Indirect I/O Transactions

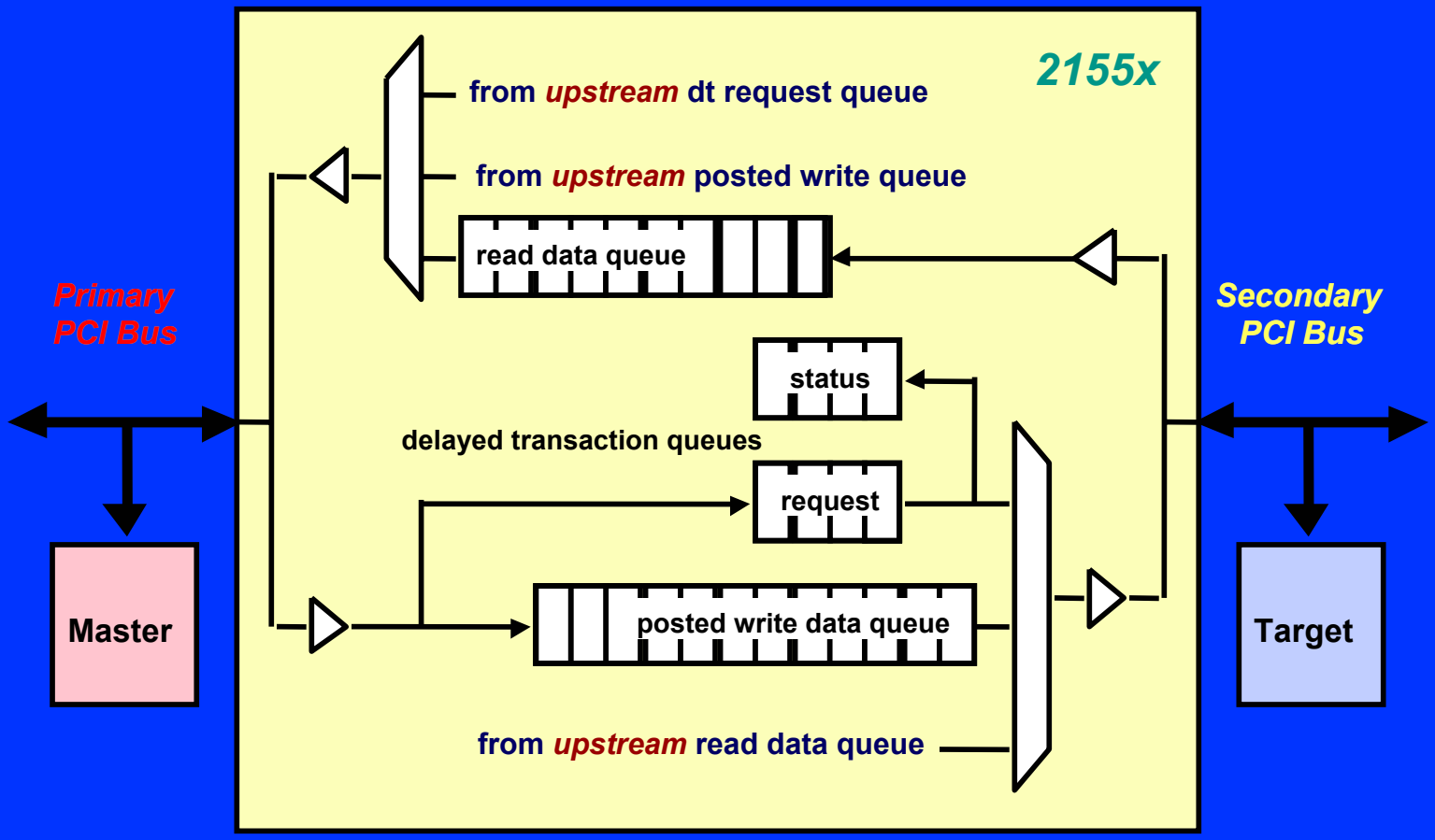
- *2155x* supports CSR initiated I/O transactions
  - provides maximum I/O addressing flexibility
  - both upstream and downstream capability
- I/O Address and I/O Data registers
  - registers are accessed in I/O mapped CSR space
  - the address is written in the I/O Address register
  - a read or write of the I/O Data register initiates the transaction
  - the transaction is queued as a delayed transaction in *2155x*

Indirect I/O Transactions



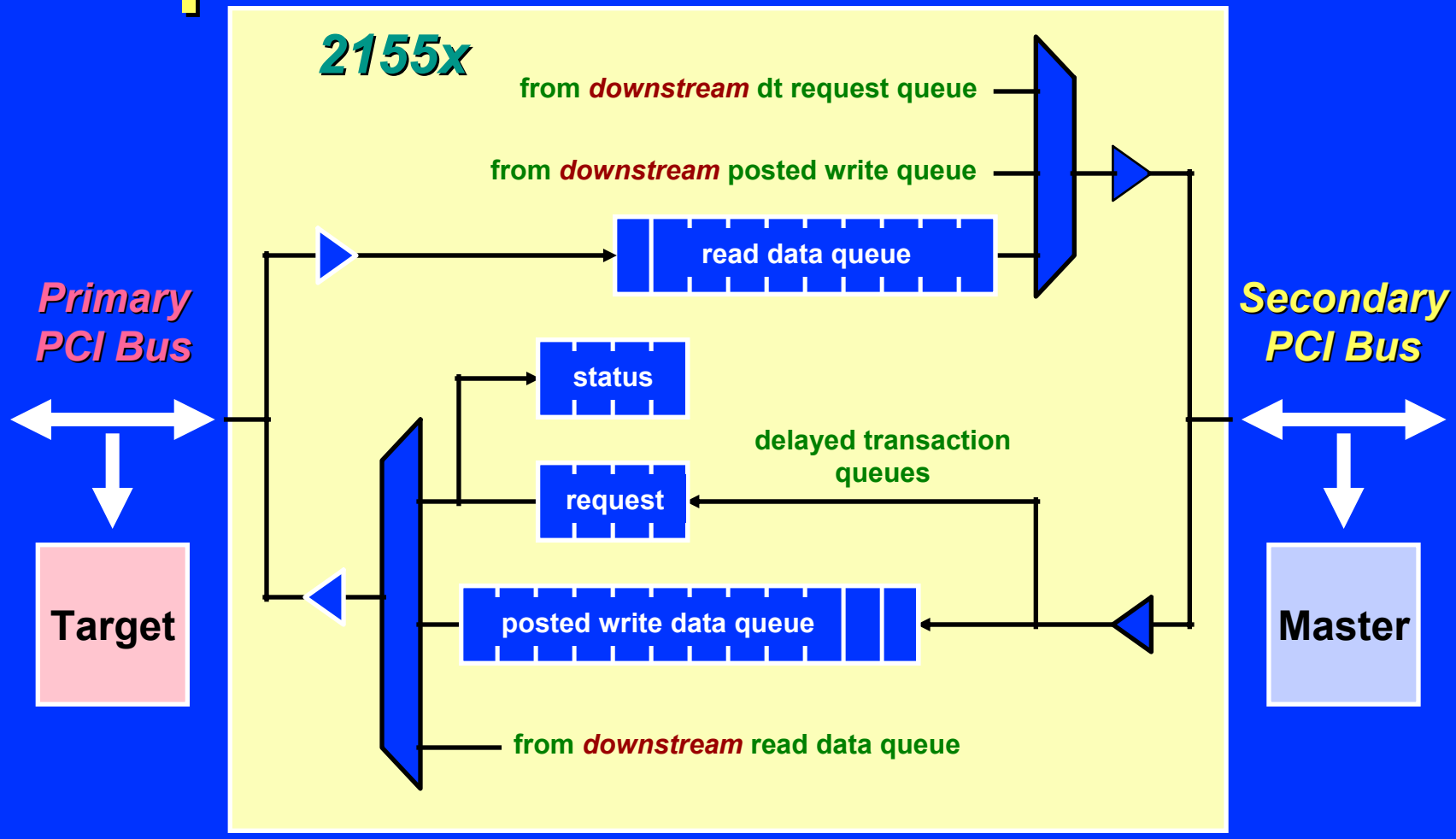
Intel Labs

Downstream Datapath



Intel Labs

Upstream Datapath



Intel Labs

Doorbells and Mailboxes

- *2155x* provides primary and secondary bus doorbell interrupts for inter-processor communication
- 16 primary and 16 secondary interrupts
- Provides eight 32-bit scratch pad registers

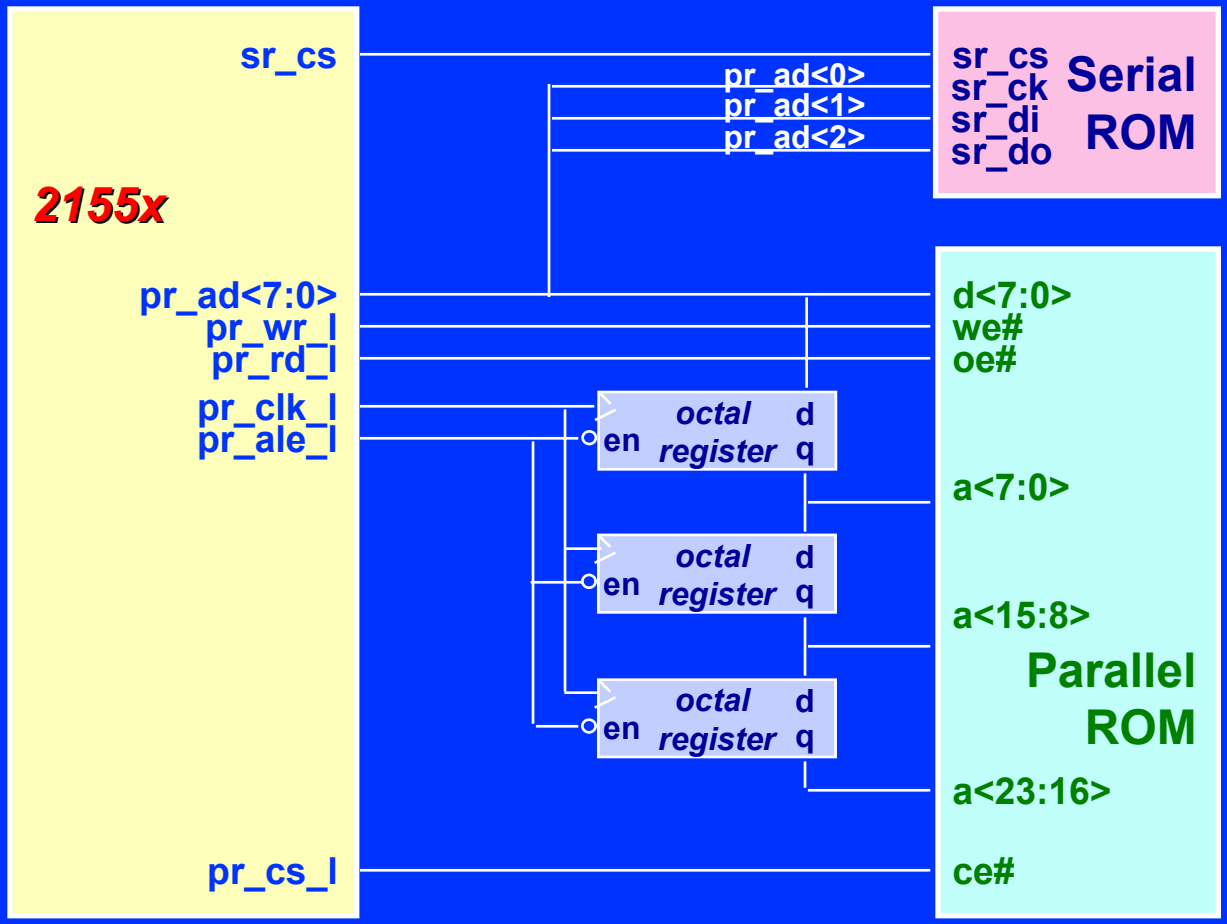
Intel Labs

I_2O® Message Unit

- *2155x* provides hardware support for the host interface to an I_2O message unit
- Inbound and outbound FIFO support
  - Host access at offsets 40h and 44h in primary memory CSR space
  - Actual queues located in local memory
  - Software manages local processor access of the queues

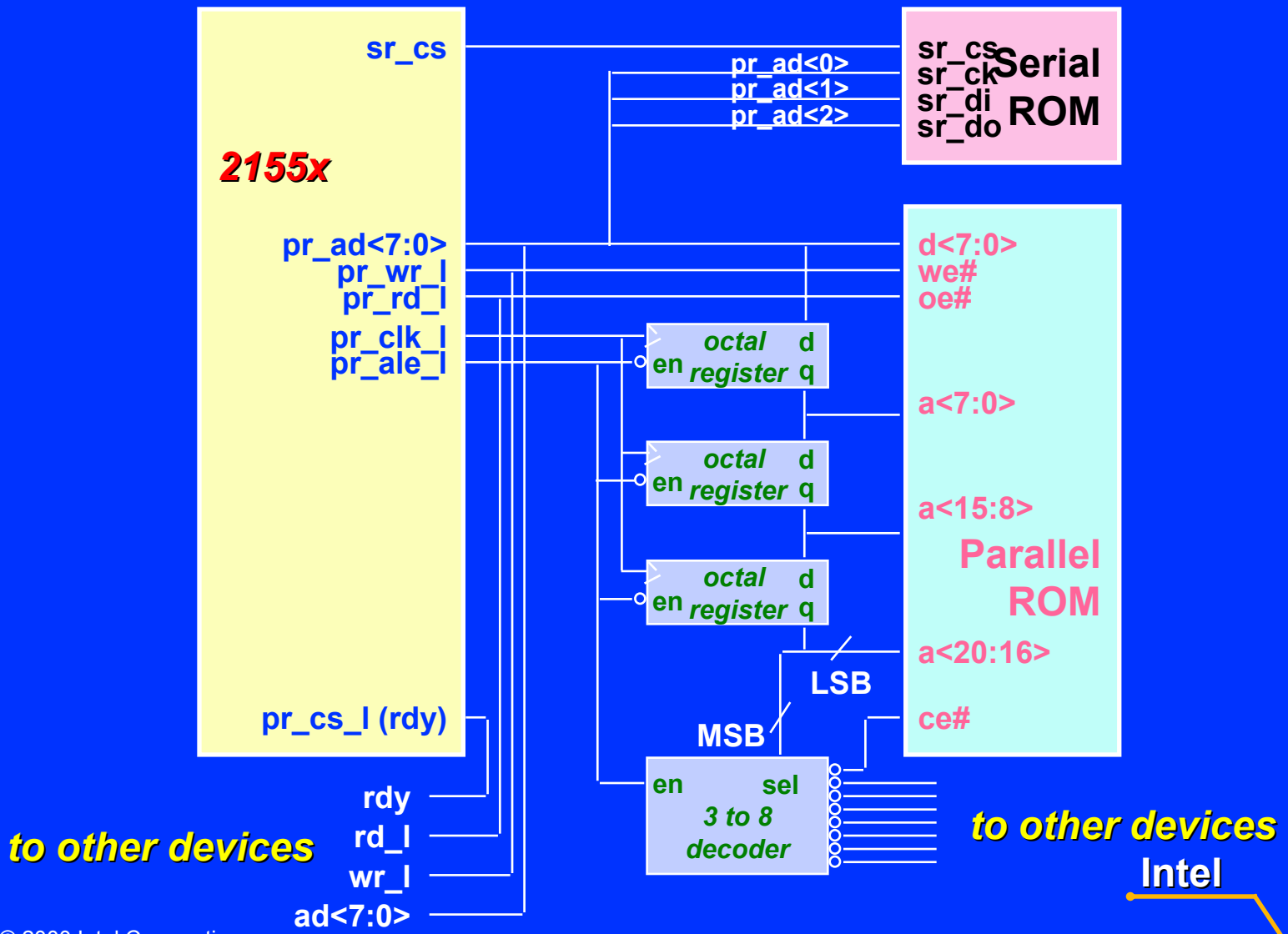
Intel Labs

Serial and Parallel ROM Connection



Intel Labs

Serial and Parallel ROM Connection



Intel Labs

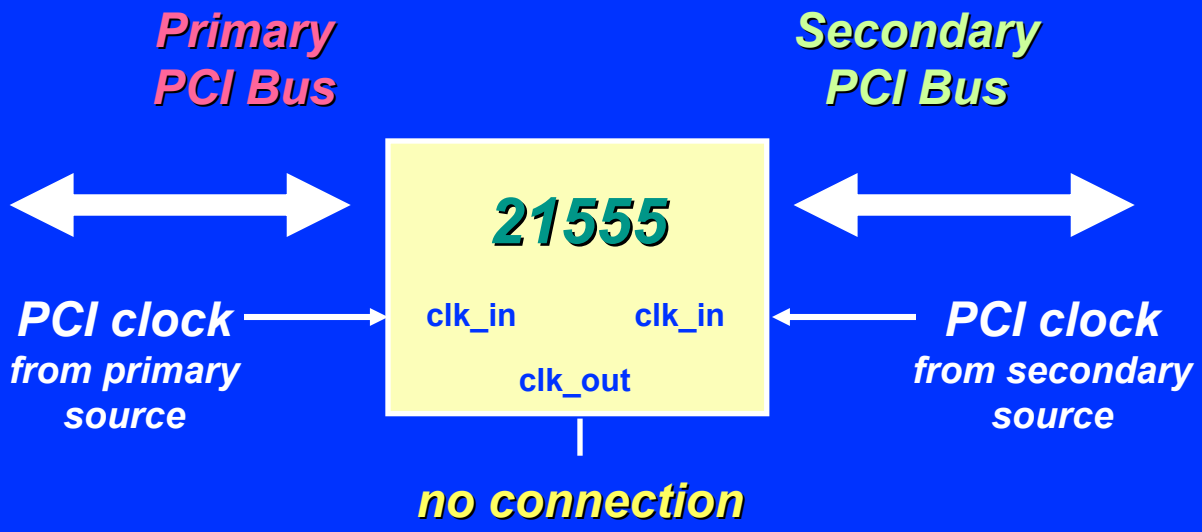
Serial ROM Access

- Following PCI reset
  - Serial ROM is read automatically to pre-load registers
- Indirect serial ROM access
  - Read, write and erase operations setup by writes to *2155x* CSRs
  - Operation initiated by setting the serial ROM start / busy bit
  - Serial interface operation is complete when start / busy bit is cleared by *2155x*
  - Some operations require poll operation to determine SROM operation status

Parallel ROM Access

- Optional sub-system PCI expansion ROM
  - Primary expansion ROM base address register (BAR)
- Indirect parallel ROM access
  - Operation setup via writes to the NTB's CSRs
  - Operation initiated by setting the parallel ROM start / busy bit
  - ROM operation is complete when start / busy bit is cleared by *2155x*

Asynchronous Clocking

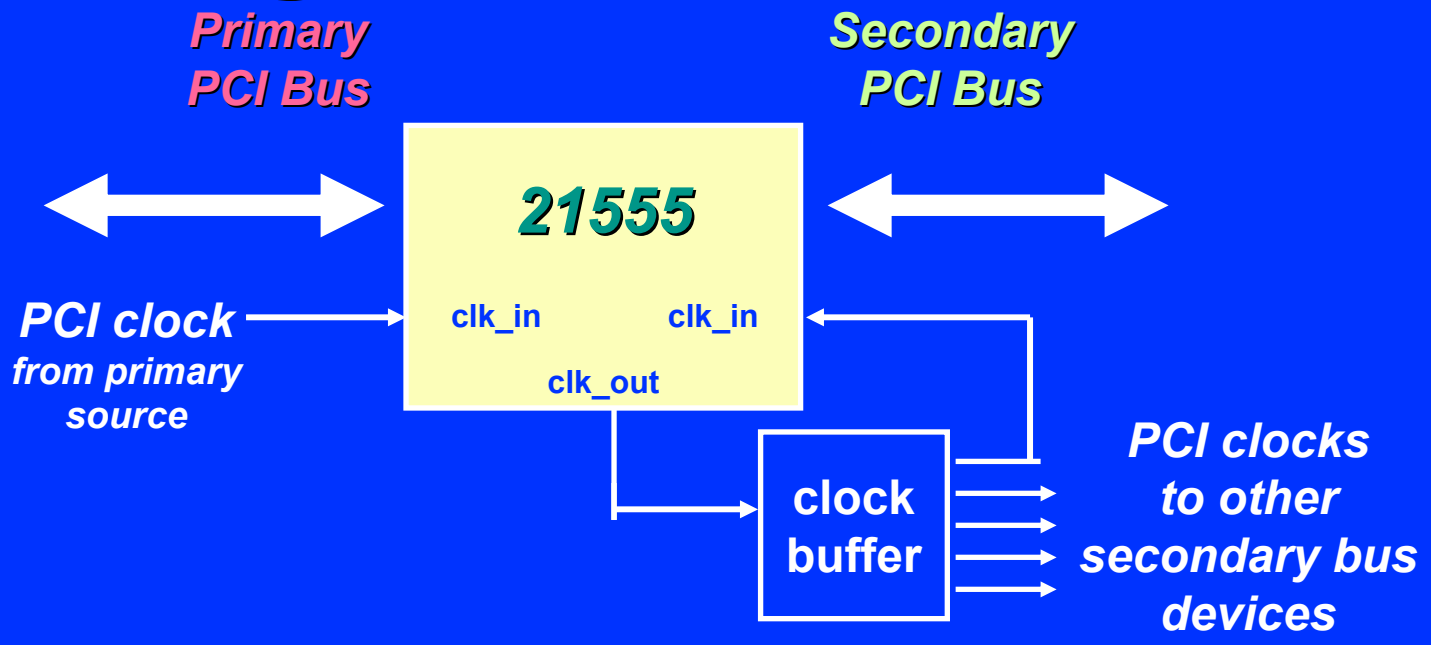


Primary and secondary clock inputs are independent

- Clock output is disabled
- Supports up to a 2.5 to 1 clock ratio

Intel Labs

Synchronous Clocking



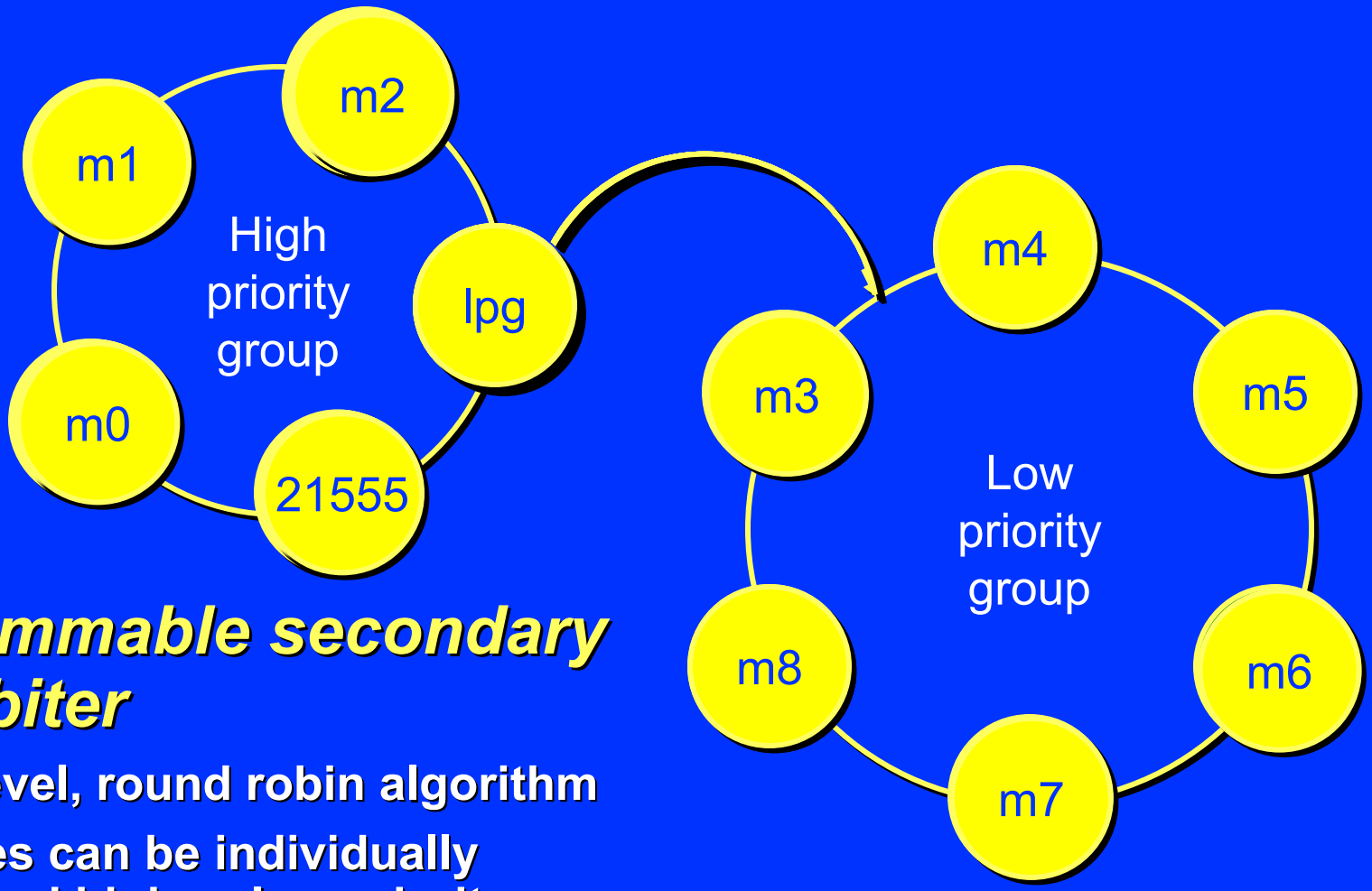
Secondary clock is derived from buffered primary clock

– Clock output is enabled

External clock buffer required

Intel Labs

Internal Arbiter



- ***Programmable secondary bus arbiter***
  - **Two level, round robin algorithm**
  - **Devices can be individually assigned high or low priority**
  - **Support for 10 bus masters including the *2155x***

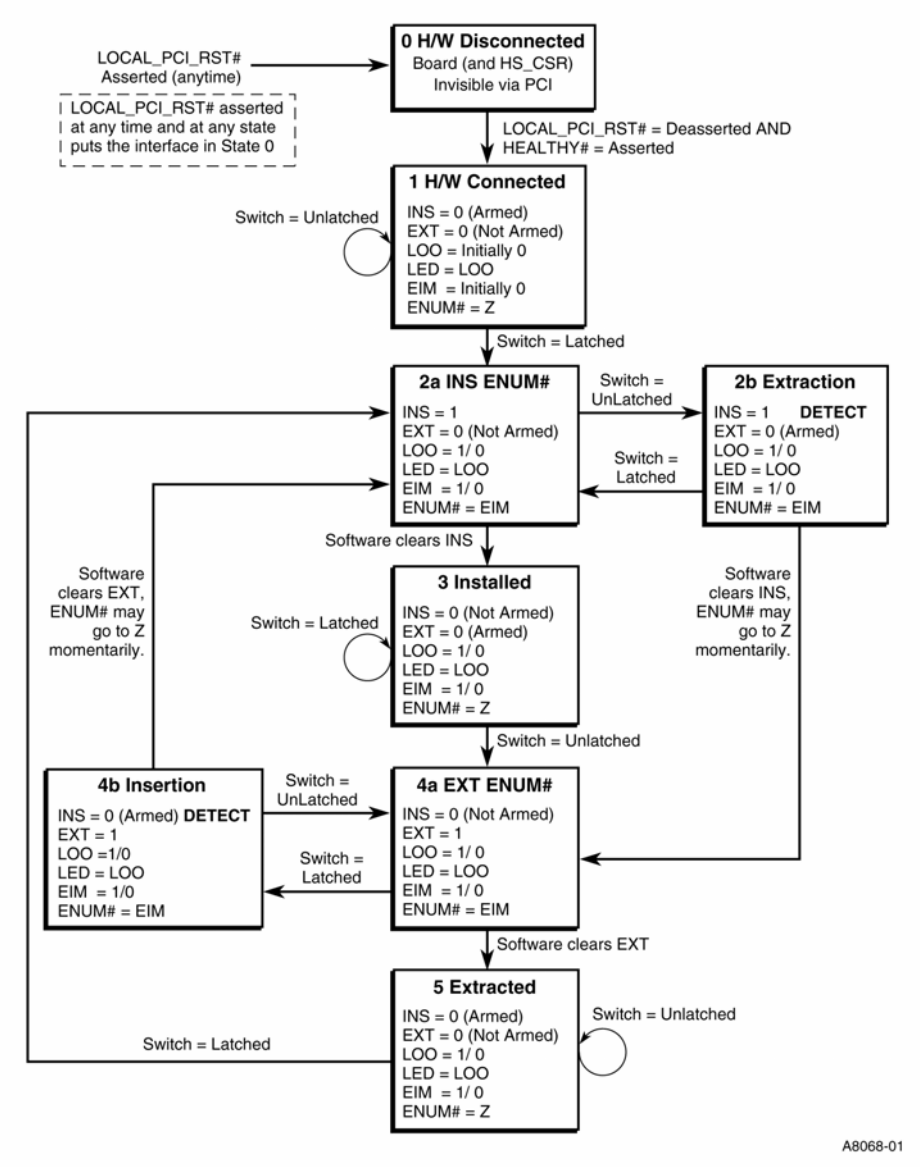
Intel Labs

21555 New Features

- 2^{24} retry counter disable
- Feature strap read-only register
- LUT page size extension bit
- Additional reset input
- Bus parking control
- Minimum GNT# assertion
- Generic memory-mapped own bits
- Auto-sense mode for hot swap signal I_stat
- Indirect configuration transactions

Intel Labs

21555 Hot Swap



Intel Labs

Summary

- *21555* has features specifically designed to meet embedded application needs
  - Flexible configuration methods
  - Flexible addressing capability
  - Flexible clocking mechanisms
  - Utilizes Intel's PCI to PCI bridge expertise
  - Uses proven rev 2.2 PPB advanced buffering technology

Appendix

21555 Non-Transparent PPB

- 304 PBGA
- 66mhz
- 64-bit primary to 64-bit secondary interface
- 256 bytes read and 256 bytes write buffering
- 4 delayed transactions each side
- PCI revision 2.2 compliant
- 3.3V operation with 5V tolerant I/O

Intel Labs

Transparent PPB
Transaction Forwarding Issues

- Addresses on secondary PCI bus are assigned by host

  Host address assignments may conflict with the needs of the local CPU

  – Address translation is needed to eliminate conflicts

- All secondary bus devices can be accessed from the primary interface

  Issue: subsystem resources may need to be isolated from host access

  – Accessible only by the local CPU or other secondary bus devices

Intel Labs

Transparent PPB Transaction Forwarding

- Transaction forwarding is based on address ranges
  - Each range is specified by a base and limit register
    - Range is not required to have a power of two (2^N) size
    - Range is not required to be naturally aligned
  - Specifies the address range for ***primary*** to ***secondary*** forwarding
    - Any address in the defined range is forwarded ***downstream***
    - ***All*** devices on secondary bus ***must*** be located in defined range

Intel Labs

Transparent PPB Transaction Forwarding continued...

- Transaction forwarding is based on address ranges
  - Inversely specifies the address range for *secondary* to *primary* forwarding
    - *Any* address not in defined range is forwarded *upstream*
  - Primary and secondary addresses are equivalent
    - *No* address translation between PCI buses

Non-Transparent PPB
Downstream Transaction Forwarding

- Transaction forwarding is based on address ranges
  - Each range is specified by a base address register (BAR)
    - Specifies the size (2^N) of the range
      - Determined by number of low order bits that are read only zero
    - Range must be naturally aligned
  - Specifies the range for *primary* to *secondary* forwarding
    - Any address in the defined range is forwarded *downstream*
    - Inverse decoding is ***NOT*** used
      - Addresses outside of translated secondary range are ***NOT*** forwarded *upstream*
      - Secondary bus devices ***can*** be isolated from host access

Intel Labs

Non-Transparent PPB
Downstream Transaction Forwarding

- Transaction forwarding is based on address ranges
  - Primary addresses ***can*** be translated
    - Direct address offset ***may*** be specified between the primary and secondary
    - Secondary bus devices to be accessed by the host ***must*** be located in ***translated*** secondary bus address range

Non-Transparent PPB
Upstream Transaction Forwarding:

- Transaction forwarding is based on address ranges
  Ranges are specified by base address registers (BAR)
  - Specifies the size (2^N) of the range
    - Determined by number of low order bits that are read only zero
  - Range must be naturally aligned
  - Specifies the range for *secondary* to *primary* forwarding
  - Any address in the defined range is forwarded *upstream*

Non-Transparent PPB Upstream Transaction Forwarding:

- Transaction forwarding is based on address ranges
  - Two methods for translating secondary addresses
    - Primary bus devices to be accessed by the host ***must*** be located in ***translated*** primary bus address range
    - Direct offset address translation ***may*** be specified between the primary and secondary bus
    - Look up table translation method for one memory address range

Intel Labs

LUT Management

- **Software support needed (local CPU)**
  - **Upon *end-of-page* interrupt**
    - **Update LUT entry**
    - **Release *end-of-page* interrupt**
      - Write 1 to clear register
  - **Service latency depends on memory page size of host CPU, PCI bus width, PCI bus frequency, allocation of LUT entries**
    - **Assuming 4K page size**
    - **Assuming 8 bus masters with 8 LUT entries allocated to each**
      - 32 bit PCI @ 33 MHz - *246 us*
      - 32 bit PCI @ 66 MHz - *143 us*
      - 64 bit PCI @ 33 MHz - *143 us*
      - 64 bit PCI @ 66 MHz - *61 us*

Intel Labs

LUT Management

- Hardware support
  - One maskable *end-of-page* interrupt per LUT entry
    - Set upon completion of last DWORD transfer within LUT entry

Intel Labs

2155x Buffer Architecture

- Uses proven buffer architecture of 2115x family
  - PCI revision 2.2 compliant
  - Multiple buffers make efficient use of bus bandwidth
    - Less time spent retrying due to buffer full conditions or transaction ordering requirements
  - Better traffic/latency isolation between bus segments
    - Non-posted transactions progress without holding the initiating bus in wait states while the target bus is being requested
  - Eliminated wait states can be used by other bus masters
    - Reduces latency and improves bus utilization efficiency

Intel Labs

ROM Interfaces

- Parallel ROM port
  - 8 bit multiplexed address/data bus
    - Up to a 24-bit address possible with external registers
  - Read, write, chip select, and address strobes provided
  - Alternate mode supports attachment of other devices
    - External chip select decoder required
    - Software configurable strobe timing (one setting used for ***all*** devices) Chip select duration and position of read/write strobes
- Serial ROM port
  - Serial ROM attaches directly to ***2155x*** (no external logic)
    - Serial ROM pins are multiplexed with parallel ROM pins
  - Uses 4-pin MicroWire interface
    - sr_cs, sr_ck, sr_di, sr_do

Clocking

- Two clock modes
  - PCI interfaces can be either *asynchronous* or *synchronous*
    - Asynchronous operation: Local CPU needs to be the secondary PCI bus clock source
    - Synchronous operation: Local CPU clock is independent from secondary bus clock source
  - Selected by mode pin during reset
- Three signal pins for clocking
  - Primary PCI clock input
  - Secondary PCI clock output
    - Buffered version of primary PCI clock
    - Can be externally buffered for distribution
  - Secondary PCI clock input

Non-Transparent PPB SROM Initialization

- Following PCI reset the *2155x's* initial state is loaded from a serial ROM
  - Vendor specific identification
    - Subsystem ID, subsystem vendor ID, class code, sub-class code, etc.
  - Base address register setup information
    - Number, size and type of address ranges
  - Initialization status bit
    - Locks out host from access of configuration registers
    - 1 - initialization must be completed by local CPU
    - 0 - initialization completed. Host now able to access configuration registers

Intel Labs

Indirect Configuration Transactions

- *2155x* does *not* support *Transparent PCI-to-PCI bridge* hierarchical configuration mechanisms
  - Host configures primary bus devices
  - Local processor configures secondary bus devices
- *2155x does* provide an alternative method.
  - Can initiate type 0 or type 1 configuration transactions via CSR accesses
- CSR indirect configuration method is the same as for I/O
  - Indirect configuration address and configuration data registers
    - Mapped in both configuration space and I/O space
  - Access completed as a delayed transaction

Intel Labs

Indirect I/O Transactions

- *2155x* supports CSR initiated I/O transactions
  - provides maximum I/O addressing flexibility
  - both upstream and downstream capability
- I/O Address and I/O Data registers
  - registers are accessed in I/O mapped *2155x* CSR space
  - the address is written in the I/O Address register
  - a read or write of the I/O Data register initiates the transaction
  - the transaction is queued as a delayed transaction
    - *2155x* will return a target retry on initial attempt
    - after the I/O transaction completes on the target bus, TRDY# (and read data) is returned when initiator repeats the I/O read or write
    - semaphore bit provided
    - used to ensure atomic usage if multiple processes utilize indirect CSRs

Intel Labs

Doorbells and Mailboxes

- *2155x* provides primary and secondary bus doorbell interrupts for inter-processor communication
- 16 primary and 16 secondary interrupts
  - Controlled by write-1-to-set and write-1-to-clear CSRs
  - Enable bits for each interrupt
    - Controlled by write-1-to-set and write-1-to-clear CSRs
  - Mapped in I/O and memory CSR space
- Provides eight 32-bit scratch pad registers
  - Mapped in I/O and memory CSR space

Intel Labs

I_2O Message Unit

- *2155x* provides hardware support for the host interface to an I_2O message unit
  - Allows message passing between the host and local processor
- Inbound and outbound FIFO support
  - Host access at offsets 40h and 44h in primary memory CSR space
  - Actual queues located in local memory
    - Typically attached to local processor

Intel Labs

I_2O Message Unit cont..

- Inbound and outbound FIFO support
  - *2155x* allocates MFAs from and posts MFAs to the inbound queues on behalf of the host
  - *2155x* retrieves MFAs from and returns MFAs to the outbound queues on behalf of the host
  - Software manages local processor access of the queues

Intel Labs

Other Features

- Programmable secondary bus arbiter
  - Two level, round robin algorithm
  - Devices can be individually assigned high or low priority
  - Support for 10 bus masters including the *2155x*
- PCI universal I/O buffers
  - Allows connection to 3.3V or 5V PCI buses
    - Signaling voltage of each PCI interface is independent
    - Can function as bridge between 3.3V and 5V PCI bus segments
- 3.3 volt supply for low power consumption
- 66 MHz PCI revision 2.2 compliant

Intel Labs

21555 New Features

- 2^{24} retry counter disable
  - When disabled, the 21555 will infinitely repeat a transaction receiving target retry. Applies to all retry timers.
- Feature strap read-only register
  - Reflects the values sampled at reset on the pr_ad pins.

Intel Labs

21555 New Features

- LUT page size extension bit
  - When set, enables three more page sizes, expanding the upstream memory 2 window capability up to 2gbytes.

21555 New Features

- Additional reset input
  - Signal borrows a VDD pin from the 21554. In pin compatible applications it will be disabled. New applications can use this as a hot swap local reset, or as a way to reset the chip from the secondary interface.

21555 New Features

- Bus parking control
  - Bus parking is selectable between 21555 and last master to optimize latency based on traffic patterns.
- Minimum GNT# assertion
  - Minimum GNT# assertion of 2 cycles to enable better interoperability with some PCI devices that cannot handle single-cycle grant assertion.

Intel Labs

21555 New Features

- Generic memory-mapped own bits
  - These bits are not tied to any hardware function, but may be used as a general synchronization mechanism.

Intel Labs

21555 New Features

- Auto-sense mode for hot swap signal I_stat
  - If the software is driving I_stat, *21555* periodically tri-states the signal and samples it to detect any changes in switch state.

- Indirect configuration transactions
  - 21555 can respond to indirect configuration transactions that it generates, enabling a general bus scan using device-specific configuration software.

Intel Labs

Serial ROM Access

- Following PCI reset
  - Serial ROM is read automatically by *2155x* to pre-load registers
- Indirect serial ROM access
  - Read, write and erase operations setup by writes to the *2155x* CSRs
    - Op code, address, and ROM write data
  - Operation initiated by setting the serial ROM start/busy bit
    - Also indicates completion status
  - Serial interface operation is complete when start/busy bit is cleared by the *2155x*
    - Read data can be read via *2155x* ROM data CSR
  - Some operations require poll operation to determine SROM operation status
    - Write, erase
    - Poll operation initiated by CSR read while serial ROM start/busy bit is set

Intel Labs

Parallel ROM Access

- Optional sub-system PCI expansion ROM
  - Primary expansion ROM base address register (BAR)
    - Defines memory range for ROM read accesses from primary bus
- Indirect parallel ROM access
  - Operation setup via writes to the *2155x* CSRs
    - Address, and ROM write data
  - Operation initiated by setting the parallel ROM start/busy bit
    - Also indicates completion status
  - ROM operation is complete when start/busy bit is cleared by the *2155x*
    - Read data can be read via *21555* ROM data CSR

Intel Labs